Efficient Inexact Solvers in Numerical Optimization Methods

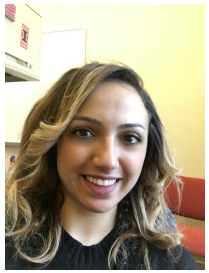
Edgar Solomonik

LPNA @CS@Illinois

Department of Computer Science
University of Illinois at Urbana-Champaign

E-NLA Seminar

Outline



Samah Karim

Linjian Ma

Navjot Singh

1. Preconditioned Solvers for Interior Point KKT Systems
2. Randomized Constrained Least Squares for Tucker Decomposition
3. Alternating Optimization for CP Decomposition

Karush-Kuhn-Tucker (KKT) conditions

Quadratic program (QP):

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & \frac{1}{2} x^T H x + x^T c \\ \text{s.t.} \quad & Ax = b, Cx \geq d \end{aligned}$$

Lagrangian function:

$$L(x, \lambda, \nu) = \frac{1}{2} x^T H x + x^T c - \lambda^T (Ax - b) - \nu^T (Cx - d)$$

First-order optimality (KKT) conditions:

$$\begin{aligned} \nabla_x L(x, \lambda, \nu) &= 0 \\ Ax - b &= 0 \\ Cx - d &\geq 0 \\ \nu^T (Cx - d) &= 0 \\ \nu &\geq 0 \end{aligned}$$

Primal-dual Interior Point Method (IPM)

Solve Perturbed KKT conditions after introducing slack variables $s \in \mathbb{R}^{m_2}$

$$\begin{aligned} Hx + c - A^T\lambda - C^T\nu &= 0 \\ Ax - b &= 0 \\ Cx - d - s &= 0 \\ SVe &= \sigma\mu e \\ s, \nu &> 0 \end{aligned}$$

where

$$V = \mathsf{diag}(\nu_1, \ldots, \nu_{m_2}), \;\; S = \mathsf{diag}(s_1, \ldots, s_{m_2}), \;\; e = [1, \ldots 1]^T \in \mathbb{R}^{m_2}$$

$$\mu = \frac{s^T\nu}{m_2}, \quad \sigma \in [0, 1]$$

Interior Point Method (IPM): KKT system

Interior point KKT equations can be written in matrix form as

$$\begin{bmatrix} -H & A^T & C^T \\ A & 0 & 0 \\ C & 0 & D^{(k)} \end{bmatrix} \begin{pmatrix} \Delta x^{(k)} \\ \Delta \lambda^{(k)} \\ \Delta \nu^{(k)} \end{pmatrix} = - \begin{pmatrix} r_g^{(k)} \\ r_e^{(k)} \\ r_a^{(k)} \end{pmatrix}$$

where $D^{(k)} = \left(V^{(k)}\right)^{-1} S^{(k)}$ is diagonal and changing with iteration k.

Traditional approach is to eliminate $\nu^{(k)}$ first, then solve iteratively

$$\begin{bmatrix} -\left(H + C^T \left(D^{(k)}\right)^{-1} C\right) & A^T \\ A & 0 \end{bmatrix} \begin{pmatrix} \Delta x^{(k)} \\ \Delta \lambda^{(k)} \end{pmatrix} = - \begin{pmatrix} r_u^{(k)} \\ r_e^{(k)} \end{pmatrix}$$

We instead use a single (for entire IPM execution) factorization of

$$F = \begin{bmatrix} -H & A^T \\ A & 0 \end{bmatrix}$$

Known Properties of IPM KKT Systems

- Iterative methods and preconditioners can be applied to both 2-by-2 and 3-by-3 systems
- Such saddle point systems are well-studied[1] and arise in numerical PDE solvers[2,3]
- Preconditioners have often been designed to exploit the block structure of the systems[4,5,6]
- The 3-by-3 system has better spectral properties, but the reduced system can nevertheless be preferable computationally[7,8]

---

[1] M. Benzi, G.H. Golub, J. Liesen. Numerical solution of saddle point problems. Acta Numerica, 2005.

[2] R. E. Ewing, R. D. Lazarov, P. Lu, P. S. Vassilevski, PCGM 1990.

[3] C. Greif, D. Schötzau, NLA 2007

[4] G.H. Golub and C. Greif, SISC 2003.

[5] C. Keller, N. I.M. Gould, and A. J. Wathen, SIMAX 2000.

[6] T. Rees, C. Greif, SISC 2007.

[7] B. Morini, V. Simoncini, M. Tani, NLA 2016.

[8] B. Morini, V. Simoncini, M. Tani, COA 2017.

Preconditioning New Reduced KKT System

At each IPM step, given a factorization of F, we iteratively solve a system with the matrix

$$K_F^{(k)} = D^{(k)} - \begin{bmatrix} C & 0 \end{bmatrix} F^{-1} \begin{bmatrix} C^T \\ 0 \end{bmatrix}$$
$$= D^{(k)} + CH^{-1}(H - \underbrace{A^T(AH^{-1}A^T)^{-1}A}_{H_A})H^{-1}C^T$$

For $A^T = QR$, $H_A = QQ^THQQ^T$ so

$$\text{rank}(H_A) \leq m_1, \quad \text{rank}(H - H_A) \leq 2(n - m_1)$$

where n is # of variables and m_1 is # equality constraints.

We propose 2 preconditioners for different regimes of # d.o.f. $n - m_1$

	Low-d.o.f.	High-d.o.f.
$\text{rank}(H_A)$	$\approx n$	≈ 0
$\text{rank}(H - H_A)$	≈ 0	$\approx n$
preconditioner	$M_L = D^{(k)}$	$M_H = D^{(k)} + CH^{-1}C^T$

Spectral Properties of Preconditioned Systems

With the low-d.o.f. preconditioner $M_L = D^{(k)}$, the preconditioned system is

$$M_L^{-1/2} K_F^{(k)} M_L^{-1/2} = I + M_L^{-1/2} C H^{-1} (H - H_A) H^{-1} C^T M_L^{-1/2}$$

- the ill-conditioned part of K_F, $D^{(k)}$, is transformed to I
- since the second term is rank $2(n - m_1)$, we have $2(n - m_1)$ non-unit eigenvalues
- CG will then converge in $2(n - m_1) + 1$ iterations

With the high-d.o.f. preconditioner $M_H = D^{(k)} + C H^{-1} C^T$

$$M_H^{-1/2} K_F^{(k)} M_H^{-1/2} = I + M_H^{-1/2} C H^{-1} H_A H^{-1} C^T M_H^{-1/2}$$

- the ill-conditioned part of K_F, $D^{(k)}$, is transformed to I
- since the second term is rank m_1, we have m_1 non-unit eigenvalues
- CG will then converge in $m_1 + 1$ iterations

Comparison to Existing Approaches

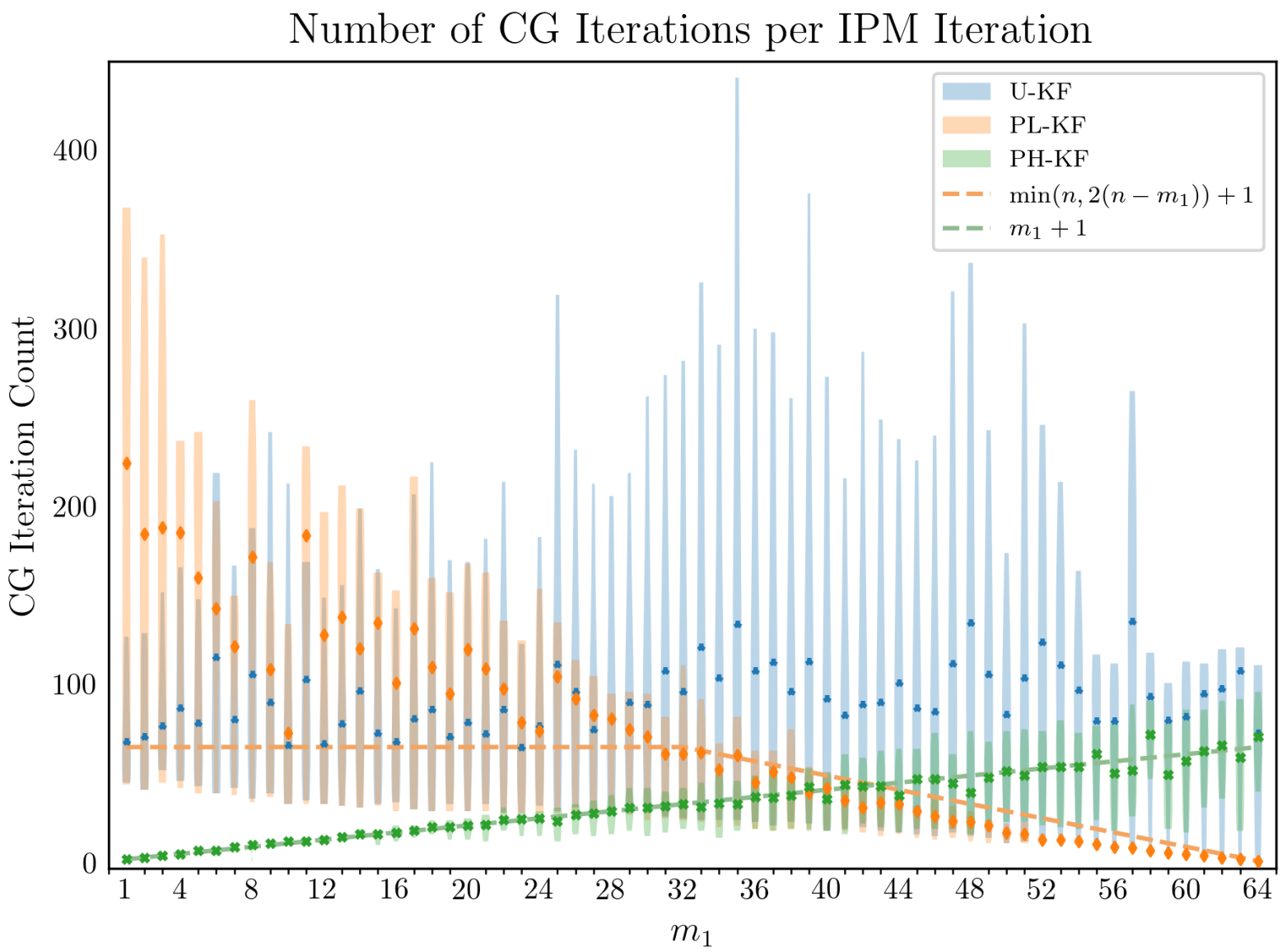
- Factorize $F = \begin{bmatrix} -H & A^T \\ A & 0 \end{bmatrix}$
- for $k = 1$ until IPM converges
  - Construct preconditioner M to be M_L or M_H depending on # d.o.f. $n - m_1$
  - Factorize M
  - Iteratively solve $M^{-1} K_F x = M^{-1} b$, by applying $K_F = D^{(k)} - \begin{bmatrix} C & 0 \end{bmatrix} F^{-1} \begin{bmatrix} C^T \\ 0 \end{bmatrix}$ in implicit form using factorization of F

- for $k = 1$ until IPM converges
  - Form augmented system $K_D = \begin{bmatrix} -\left(H + C^T \left(D^{(k)}\right)^{-1} C\right) & A^T \\ A & 0 \end{bmatrix}$
  - Choose M among preconditioners, e.g., constraint preconditioner $\begin{bmatrix} \tilde{D}^{(k)} & A^T \\ A & 0 \end{bmatrix}$ or block-diagonal $\begin{bmatrix} \tilde{D}^{(k)} - A^T W^{(k)} A & \\ & \gamma I \end{bmatrix}$
  - Factorize M
  - Iteratively solve $M^{-1} K_D x = M^{-1} b$

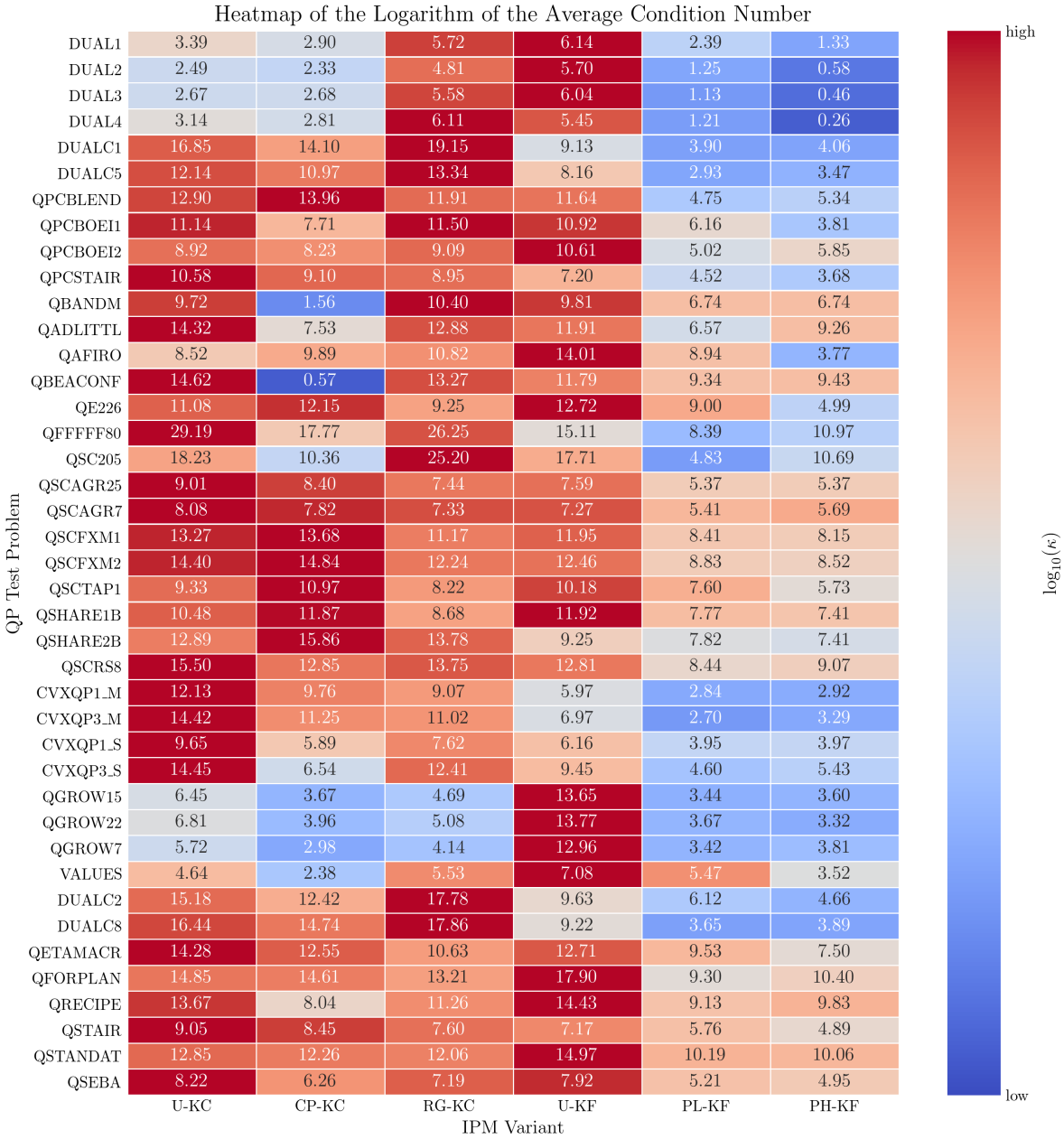
Limitations

- What if H is semidefinite? What if F is singular?
  - our approach assumed we can factorize F
  - our high-d.o.f. preconditioner $M_H = D^{(k)} + CH^{-1}C^T$ assumed H is nonsingular
  - regularization can ensure H and F are nonsingular
- When H is semidefinite but F is nonsingular
  - can factorize F with pivoting, but need more complicated high-d.o.f. preconditioner
- When H and F are singular (A is assumed to be full rank)
  - singularity often comes from zero blocks in H associated with linear variables in QP
  - can pivot linear variables and other null space to 'end' (unfactorized part of system)
- How expensive are factorization and subsequent solves with F?

CG Convergence Results

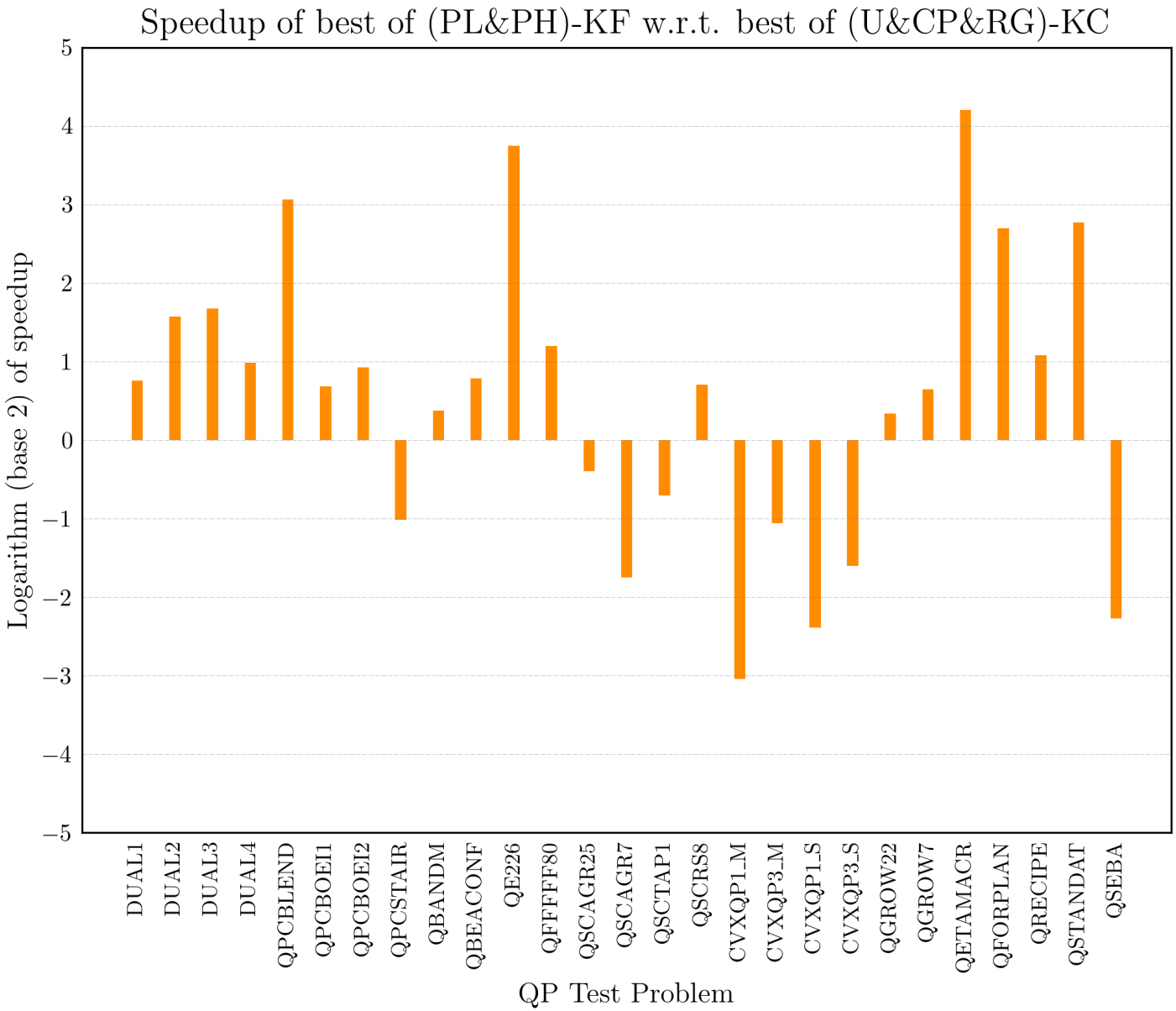


Condition Number Improvement



QP Test Problem	U-KC	CP-KC	RG-KC	U-KF	PL-KF	PH-KF
DUAL1	3.39	2.90	5.72	6.14	2.39	1.33
DUAL2	2.49	2.33	4.81	5.70	1.25	0.58
DUAL3	2.67	2.68	5.58	6.04	1.13	0.46
DUAL4	3.14	2.81	6.11	5.45	1.21	0.26
DUALC1	16.85	14.10	19.15	9.13	3.90	4.06
DUALC5	12.14	10.97	13.34	8.16	2.93	3.47
QPCBLEND	12.90	13.96	11.91	11.64	4.75	5.34
QPCBOEI1	11.14	7.71	11.50	10.92	6.16	3.81
QPCBOEI2	8.92	8.23	9.09	10.61	5.02	5.85
QPCSTAIR	10.58	9.10	8.95	7.20	4.52	3.68
QBANDM	9.72	1.56	10.40	9.81	6.74	6.74
QADLITTL	14.32	7.53	12.88	11.91	6.57	9.26
QAFIRO	8.52	9.89	10.82	14.01	8.94	3.77
QBEACONF	14.62	0.57	13.27	11.79	9.34	9.43
QE226	11.08	12.15	9.25	12.72	9.00	4.99
QFFFFF80	29.19	17.77	26.25	15.11	8.39	10.97
QSC205	18.23	10.36	25.20	17.71	4.83	10.69
QSCAGR25	9.01	8.40	7.44	7.59	5.37	5.37
QSCAGR7	8.08	7.82	7.33	7.27	5.41	5.69
QSCFXM1	13.27	13.68	11.17	11.95	8.41	8.15
QSCFXM2	14.40	14.84	12.24	12.46	8.83	8.52
QSCTAP1	9.33	10.97	8.22	10.18	7.60	5.73
QSHARE1B	10.48	11.87	8.68	11.92	7.77	7.41
QSHARE2B	12.89	15.86	13.78	9.25	7.82	7.41
QSCRS8	15.50	12.85	13.75	12.81	8.44	9.07
CVXQP1_M	12.13	9.76	9.07	5.97	2.84	2.92
CVXQP3_M	14.42	11.25	11.02	6.97	2.70	3.29
CVXQP1_S	9.65	5.89	7.62	6.16	3.95	3.97
CVXQP3_S	14.45	6.54	12.41	9.45	4.60	5.43
QGROW15	6.45	3.67	4.69	13.65	3.44	3.60
QGROW22	6.81	3.96	5.08	13.77	3.67	3.32
QGROW7	5.72	2.98	4.14	12.96	3.42	3.81
VALUES	4.64	2.38	5.53	7.08	5.47	3.52
DUALC2	15.18	12.42	17.78	9.63	6.12	4.66
DUALC8	16.44	14.74	17.86	9.22	3.65	3.89
QETAMACR	14.28	12.55	10.63	12.71	9.53	7.50
QFORPLAN	14.85	14.61	13.21	17.90	9.30	10.40
QRECIPE	13.67	8.04	11.26	14.43	9.13	9.83
QSTAIR	9.05	8.45	7.60	7.17	5.76	4.89
QSTANDAT	12.85	12.26	12.06	14.97	10.19	10.06
QSEBA	8.22	6.26	7.19	7.92	5.21	4.95

Arithmetic Cost Model Comparison



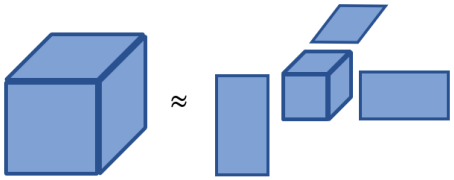
Summary and Reference

- We present a method for solving IPM KKT systems that performs a factorization of a subsystem that is fixed throughout IPM iterations
- The method is a hybrid of direct and iterative, as it requires factorizing a large sparse matrix
- The preconditioned reduced systems we solve are well-conditioned in theory and numerical results
- For more details, see "Efficient Preconditioners for Interior Point Methods via a New Schur Complementation Strategy", Samah Karim, E.S., arXiv:2104.12916

Background on Tensor Decompositions

Tucker decomposition

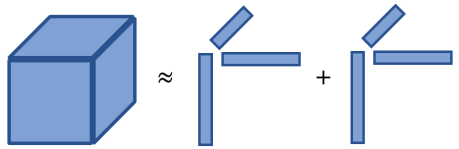
$$\mathcal{T} \approx \mathcal{X} \times_1 \boldsymbol{A} \times_2 \boldsymbol{B} \times_3 \boldsymbol{C}$$



- $\mathcal{T} \in \mathbb{R}^{s \times s \times s}$, $\mathcal{X} \in \mathbb{R}^{R \times R \times R}$
- $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C} \in \mathbb{R}^{s \times R}$ with orthonormal columns, $R < s$

CP decomposition

$$\mathcal{T} \approx \sum_{r=1}^{R} \boldsymbol{a}_r \circ \boldsymbol{b}_r \circ \boldsymbol{c}_r$$



- $\mathcal{T} \in \mathbb{R}^{s \times s \times s}$, $\boldsymbol{A} = [\boldsymbol{a}_1, \ldots, \boldsymbol{a}_R] \in \mathbb{R}^{s \times R}$
- $R < s^2$

Higher order orthogonal iteration (HOOI)

$$\min_{\boldsymbol{A}, \mathcal{X}} \frac{1}{2} \left\| (\boldsymbol{C} \otimes \boldsymbol{B}) \boldsymbol{X}_{(1)}^T \boldsymbol{A}^T - \boldsymbol{T}_{(1)}^T \right\|_F^2$$

CP-Alternating least squares (CP-ALS)

$$\min_{\boldsymbol{A}} \frac{1}{2} \left\| (\boldsymbol{C} \odot \boldsymbol{B}) \boldsymbol{A}^T - \boldsymbol{T}_{(1)}^T \right\|_F^2$$

Background

Higher order orthogonal iteration (HOOI)

$$\min_{\boldsymbol{A},\boldsymbol{\mathcal{X}}} \frac{1}{2} \left\| (\boldsymbol{C} \otimes \boldsymbol{B}) \boldsymbol{X}_{(1)}^T \boldsymbol{A}^T - \boldsymbol{T}_{(1)}^T \right\|_F^2$$

- Kronecker product $\boldsymbol{C} \otimes \boldsymbol{B} \in \mathbb{R}^{s^2 \times R^2}$
- Costs $\Theta(s^3 R)$ or $\Theta(\text{nnz}(\boldsymbol{\mathcal{T}}) R^2)$
- Fast convergence

CP-Alternating least squares (CP-ALS)

$$\min_{\boldsymbol{A}} \frac{1}{2} \left\| (\boldsymbol{C} \odot \boldsymbol{B}) \boldsymbol{A}^T - \boldsymbol{T}_{(1)}^T \right\|_F^2$$

- Khatri-Rao product $\boldsymbol{C} \odot \boldsymbol{B} \in \mathbb{R}^{s^2 \times R}$
- Costs $\Theta(s^3 R)$ or $\Theta(\text{nnz}(\boldsymbol{\mathcal{T}}) R)$
- Slow convergence

Low rank approximation ($R \ll s$):

- Sketched HOOI for Tucker decomposition (Linjian Ma, and E.S., arxiv 2104.01101)
- Overall cost with t HOOI sweeps reduced to $O\left(\text{nnz}(\boldsymbol{\mathcal{T}}) + t\left(sR^3 + R^6\right)\right)$
- Can also accelerate CPD via performing CP-ALS on the Tucker core tensor

Sketched HOOI for Tucker decomposition

HOOI: solve and truncate

$$\min_{\boldsymbol{P}\in\mathbb{R}^{s\times R^2}} \frac{1}{2}\left\|(\boldsymbol{C}\otimes\boldsymbol{B})\boldsymbol{P}^T-\boldsymbol{T}_{(1)}^T\right\|_F^2$$

$\boldsymbol{A}\boldsymbol{X}_{(1)} \leftarrow$ Best rank-R approximation of $\boldsymbol{P}$

Sketched HOOI: sketch, solve and truncate

$$\min_{\hat{\boldsymbol{P}}\in\mathbb{R}^{s\times R^2}} \frac{1}{2}\left\|\boldsymbol{S}(\boldsymbol{C}\otimes\boldsymbol{B})\hat{\boldsymbol{P}}^T-\boldsymbol{S}\boldsymbol{T}_{(1)}^T\right\|_F^2$$

$\hat{\boldsymbol{A}}\hat{\boldsymbol{X}}_{(1)} \leftarrow$ Best rank-R approximation of $\hat{\boldsymbol{P}}$

- $\boldsymbol{S}\in\mathbb{R}^{m\times s^2}$ is the sketching matrix, $m<s^2$ is the sketch size
- Sketched **rank-constrained** linear least squares problem
- Goal: find $\boldsymbol{S}$ such that with high probability

$$\frac{1}{2}\left\|(\boldsymbol{C}\otimes\boldsymbol{B})\hat{\boldsymbol{X}}_{(1)}^T\hat{\boldsymbol{A}}^T-\boldsymbol{T}_{(1)}^T\right\|_F^2 \leq (1+O(\epsilon))\frac{1}{2}\left\|(\boldsymbol{C}\otimes\boldsymbol{B})\boldsymbol{X}_{(1)}^T\boldsymbol{A}^T-\boldsymbol{T}_{(1)}^T\right\|_F^2$$

Sketched HOOI for Tucker decomposition

Let $\boldsymbol{S} \in \mathbb{R}^{m \times s}$ be a $(1/2, \delta, \epsilon)$-accurate sketching matrix for the LHS $\boldsymbol{C} \otimes \boldsymbol{B}$. Then we have with probability at least $1 - \delta$,

$$\frac{1}{2} \left\| (\boldsymbol{C} \otimes \boldsymbol{B}) \hat{\boldsymbol{X}}_{(1)}^T \hat{\boldsymbol{A}}^T - \boldsymbol{T}_{(1)}^T \right\|_F^2 \leq (1 + O(\epsilon)) \frac{1}{2} \left\| (\boldsymbol{C} \otimes \boldsymbol{B}) \boldsymbol{X}_{(1)}^T \boldsymbol{A}^T - \boldsymbol{T}_{(1)}^T \right\|_F^2.$$

Sketching matrices satisfying the $(1/2, \delta, \epsilon)$-accurate property

- TensorSketch (R. Pagh, TOCT 2013) with $m = O\left(R^2/\delta \cdot (R^2 + 1/\epsilon^2)\right)$
- Leverage score sampling with $m = O\left(R^2/(\epsilon^2 \delta)\right)$
- Sketch size upper bounds are at most $O(1/\epsilon)$ times the upper bounds for unconstrained linear least squares problem

Cost comparison for order 3 tensor

ALS + TensorSketch (Malik and Becker, NeurIPS 2018)

- Solving for each factor matrix or the core tensor at a time
- $\min_{\boldsymbol{A}} \frac{1}{2} \left\| (\boldsymbol{C} \otimes \boldsymbol{B}) \boldsymbol{X}_{(1)}^T \boldsymbol{A}^T - \boldsymbol{T}_{(1)}^T \right\|_F^2$ or
  $\min_{\boldsymbol{\mathcal{X}}} \frac{1}{2} \| (\boldsymbol{C} \otimes \boldsymbol{B} \otimes \boldsymbol{A}) \mathrm{vec}(\boldsymbol{X}) - \mathrm{vec}(\boldsymbol{T}) \|_F^2$

Algorithm for Tucker	LS subproblem cost	Sketch size (m)
HOOI	$\Omega(\mathrm{nnz}(\boldsymbol{\mathcal{T}})R)$	/
ALS + TensorSketch	$\tilde{O}(msR + mR^3)$	$O(R^2/\delta \cdot (R^2 + 1/\epsilon))$
HOOI + TensorSketch	$O(msR + mR^4)$	$O(R^2/\delta \cdot (R^2 + 1/\epsilon^2))$
HOOI + leverage scores	$O(msR + mR^4)$	$O(R^2/(\epsilon^2\delta))$

Sketched HOOI algorithm

Input: Input order N tensor $\mathcal{T}$, Tucker rank R, number of sweeps I_{max}, tolerance ϵ

Output: $\{\mathcal{X}, \boldsymbol{A}^{(1)}, \ldots, \boldsymbol{A}^{(N)}\}$

For $n \in \{2, ..., N\}$ **do**

$\boldsymbol{A}^{(n)} \leftarrow$ `Init-RRF`$(\boldsymbol{T}_{(n)}, R, \epsilon)$ // Initialize with randomized range finder

Endfor

For $i \in \{1, ..., I_{max}\}$ **do**

For $n \in \{1, ..., N\}$ **do**

Build the sketching matrix $\boldsymbol{S}$

$\boldsymbol{Y} \leftarrow \boldsymbol{S}\boldsymbol{T}_{(n)}$

$\boldsymbol{Z} \leftarrow \boldsymbol{S}^{(n)}(\boldsymbol{A}^{(1)} \otimes \cdots \otimes \boldsymbol{A}^{(n-1)} \otimes \boldsymbol{A}^{(n+1)} \otimes \cdots \otimes \boldsymbol{A}^{(N)})$

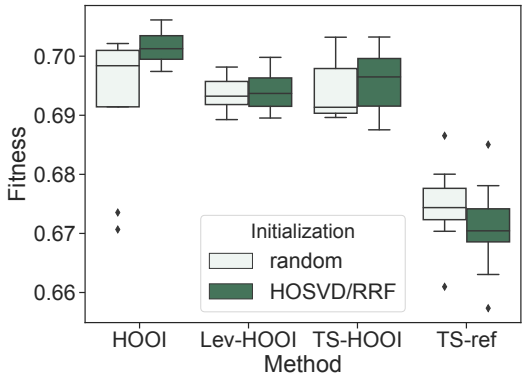
$\boldsymbol{X}_{(n)}^{T}, \boldsymbol{A}^{(n)} \leftarrow$ `Solve-truncate`$(\boldsymbol{Z}, \boldsymbol{Y}, R)$

Endfor

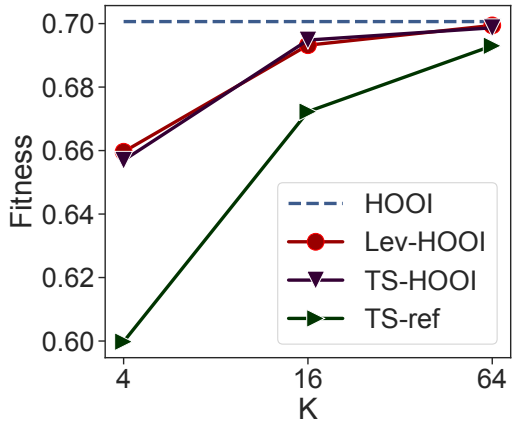
Endfor

Return $\{\mathcal{X}, \boldsymbol{A}^{(1)}, \ldots, \boldsymbol{A}^{(N)}\}$

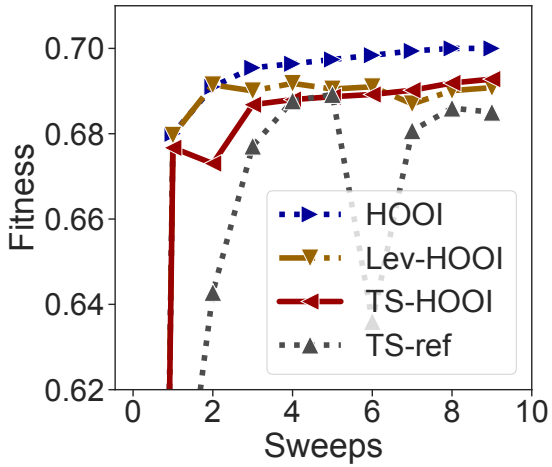
Experiments: tensors with spiked signal



(a) 5 sweeps, sample size $16R^2$



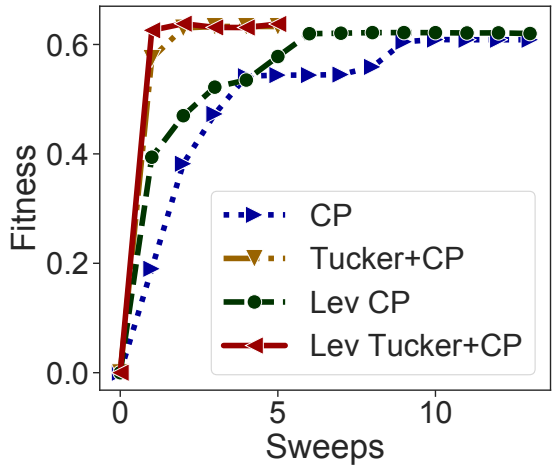
(b) 5 sweeps, sample size KR^2



(c) sample size $16R^2$

- $\boldsymbol{\mathcal{T}} = \boldsymbol{\mathcal{T}}_0 + \sum_{i=1}^{5} \lambda_i \boldsymbol{a}_i \circ \boldsymbol{b}_i \circ \boldsymbol{c}_i$, each $\boldsymbol{a}_i, \boldsymbol{b}_i, \boldsymbol{c}_i$ has unit 2-norm, $\lambda_i = 3\frac{\|\boldsymbol{\mathcal{T}}_0\|_F}{i^{1.5}}$
- Leading low-rank components obey the power-law distribution
- Tensor size $200 \times 200 \times 200$, $R = 5$
- TS-ref: (Malik and Becker, NeurIPS 2018)

Experiments: CP decomposition



- $\boldsymbol{\mathcal{T}} = \sum_{i=1}^{R_{\text{true}}} \boldsymbol{a}_i \circ \boldsymbol{b}_i \circ \boldsymbol{c}_i$, $R_{\text{true}}/R = 1.2$
- Tensor size $2000 \times 2000 \times 2000$, $R = 10$, sample size $16R^2$
- Lev CP: leverage score sampling for CP-ALS (Larsen and Kolda, arXiv:2006.16438)
- Tucker+CP: Run Tucker HOOI first, then run CP-ALS on the Tucker core
- Run Tucker HOOI with 5 sweeps, CP-ALS with 25 sweeps

New Alternating Update Scheme for CPD

A rank R CP decomposition of an $s \times s \times s$ tensor $\boldsymbol{\mathcal{X}}$,

$$\boldsymbol{\mathcal{X}} = [\![\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}]\!], \quad \text{i.e.,} \quad x_{ijk} = \sum_{r=1}^{R} a_{ir} b_{jr} c_{kr}$$

is obtained by ALS via successively minimizing

$$f(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}) = \|\boldsymbol{\mathcal{X}} - [\![\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}]\!]\|_F$$

by alternating updates such as

$$\boldsymbol{A} = \boldsymbol{\mathcal{T}}_{(1)} (\boldsymbol{C} \odot \boldsymbol{B})^{\dagger T}$$

We propose a different update, which for $R \leq s$ is

$$\boldsymbol{A} = \boldsymbol{\mathcal{T}}_{(1)} (\boldsymbol{C}^{\dagger T} \odot \boldsymbol{B}^{\dagger T})$$

Convergence to Exact Decomposition

When seeking an exact decomposition for a rank $R \leq s$ tensor

- ALS achieves a linear convergence rate[1]
- High-order convergence possible via optimizing all variables via Gauss-Newton,[2,3,4] but is costly per iteration relative to ALS
- The proposed algorithm achieves at least quartic order local convergence per sweep of alternating updates
  - per alternating update, convergence order is α where α is the positive root of $x^{N-1} - \sum_{i=0}^{N-2} x^i$ for order N tensor
  - cost per iteration is roughly the same as ALS (dominated by single matricized tensor times Khatri-Rao product (MTTKRP))

[1] A. Uschmajew, SIMAX 2012

[2] P. Paatero, Chemometrics and Intelligent Laboratory Systems 1997.

[3] A.H. Phan, P. Tichavsky, A. Cichocki, SIMAX 2013.

[4] N. Singh, L. Ma, H. Yang, E.S., SISC 2021.

Exact Decomposition Error Analysis

The error in one factor scales with the product of the errors in the others.

Lemma

Suppose $\boldsymbol{\mathcal{X}} = [\![\boldsymbol{A}^{(1)}, \ldots, \boldsymbol{A}^{(N)}]\!]$, *where each* $\boldsymbol{A}^{(i)} \in \mathbb{R}^{s_i \times R}$ *is full rank, each* $s_i \geq R$, *and* $\tilde{\boldsymbol{A}}^{(n)} \boldsymbol{D}^{(n)} = \boldsymbol{A}^{(n)} + \boldsymbol{\Delta}^{(n)}$ *for some set of diagonal matrices* $\boldsymbol{D}^{(n)} \in \mathbb{R}^{R \times R}$, *for* $n = 2, \ldots, N$, *then for sufficiently* ϵ_n,

$$\tilde{\boldsymbol{A}}^{(1)} = \boldsymbol{\mathcal{X}}_{(1)} (\tilde{\boldsymbol{A}}^{(N)\dagger T} \odot \cdots \odot \tilde{\boldsymbol{A}}^{(N)\dagger T})$$

satisfies

$$\| \tilde{\boldsymbol{A}}^{(1)} \boldsymbol{D}^{(1)} - \boldsymbol{A}^{(1)} \|_F = O(\prod_{n=2}^{N} \epsilon_n),$$

for some diagonal matrix $\boldsymbol{D}^{(1)}$.

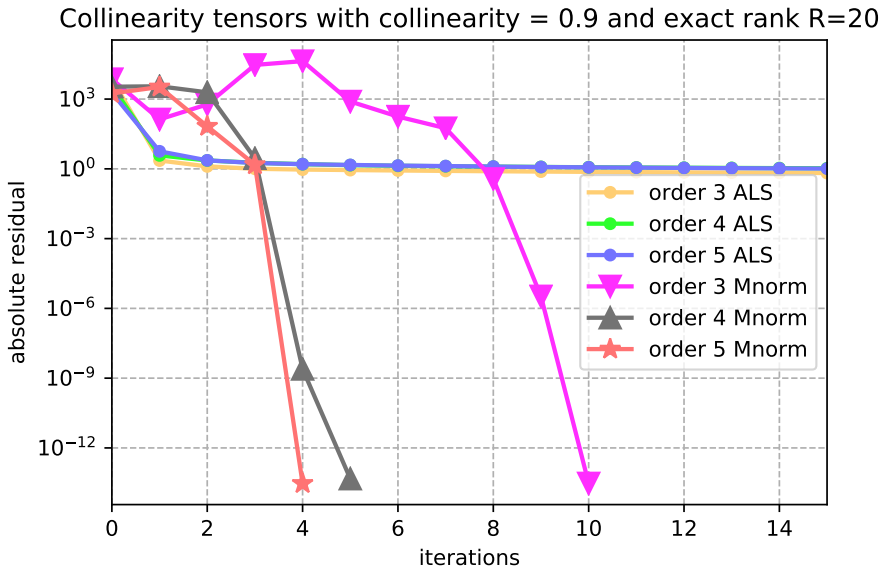
Exact Decomposition Error Analysis

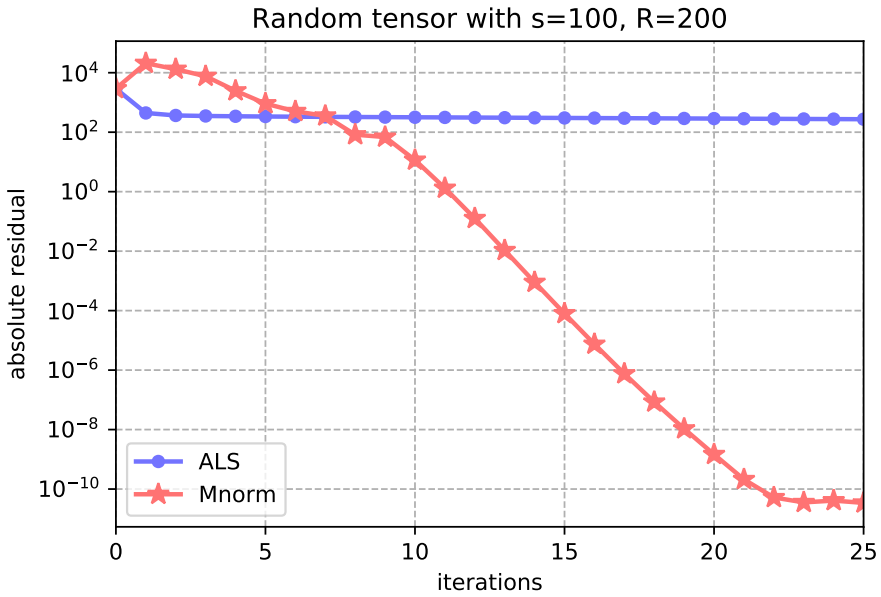
Error bound follows from substituting true decomposition into update rule

$$
\begin{aligned}
\tilde{\boldsymbol{A}}^{(1)} &= \boldsymbol{A}^{(1)} \left((\tilde{\boldsymbol{A}}^{(2)\dagger} \boldsymbol{A}^{(2)}) * \cdots * (\tilde{\boldsymbol{A}}^{(N)\dagger} \boldsymbol{A}^{(N)}) \right)^T \\
&= \boldsymbol{A}^{(1)} \left((\tilde{\boldsymbol{A}}^{(2)\dagger} (\tilde{\boldsymbol{A}}^{(2)} \boldsymbol{D}^{(2)} - \boldsymbol{\Delta}^{(2)})) * \cdots * (\tilde{\boldsymbol{A}}^{(N)\dagger} (\tilde{\boldsymbol{A}}^{(N)} \boldsymbol{D}^{(N)} - \boldsymbol{\Delta}^{(N)})) \right)^T \\
&= \boldsymbol{A}^{(1)} \left((\boldsymbol{D}^{(2)} - \tilde{\boldsymbol{A}}^{(2)\dagger} \boldsymbol{\Delta}^{(2)}) * \cdots * (\boldsymbol{D}^{(N)} - \tilde{\boldsymbol{A}}^{(N)\dagger} \boldsymbol{\Delta}^{(N)}) \right)^T \\
&= \boldsymbol{A}^{(1)} \left(\boldsymbol{D} + (-1)^{N-1} \tilde{\boldsymbol{A}}^{(2)\dagger} \boldsymbol{\Delta}^{(2)} * \cdots * \tilde{\boldsymbol{A}}^{(N)\dagger} \boldsymbol{\Delta}^{(N)} \right)^T
\end{aligned}
$$

where $\boldsymbol{D}$ is diagonal

Exact Decomposition Experimental Performance





General Fixed Points of the Update Scheme

- When the true CP rank of $\boldsymbol{\mathcal{X}}$ is greater than R, we may understand the new update scheme by looking at its fixed points
- The factors and pseudoinverses of the factors $\boldsymbol{U}^{(n)} = \boldsymbol{A}^{(n)\dagger}$ satisfy

$$\boldsymbol{A}^{(n)} = \boldsymbol{X}_{(n)}^{(N)} \left(\bigodot_{m=1, m\neq n}^{N} \boldsymbol{U}^{(m)T} \right)$$

- Premultiplying by $\boldsymbol{U}^{(n)}$, the above conditions imply

$$\boldsymbol{I} = \boldsymbol{U}^{(n)} \boldsymbol{X}_{(n)}^{(N)} \left(\bigodot_{m=1, m\neq n}^{N} \boldsymbol{U}^{(m)T} \right)$$

- We may rewrite these in terms of the multilinear function associated with the tensor

$$f_{\boldsymbol{\mathcal{X}}}(\boldsymbol{v}^{(1)}, \ldots, \boldsymbol{v}^{(N)}) = \sum_{i_1 \ldots i_N} x_{i_1 \ldots i_N} \prod_{j=1}^{N} v_{i_j}^{(j)},$$

as a set of orthonormality conditions on the rows of each $\boldsymbol{U}^{(n)}$:

$$f_{\boldsymbol{\mathcal{X}}}(\boldsymbol{u}_i^{(1)}, \ldots, \boldsymbol{u}_i^{(n-1)}, \boldsymbol{u}_j^{(n)}, \boldsymbol{u}_i^{(n+1)}, \ldots, \boldsymbol{u}_i^{(N)}) = \delta_{ij}$$

Approximate CP Decomposition

- At each step, the update $\boldsymbol{A} = \boldsymbol{\mathcal{T}}_{(1)}(\boldsymbol{C}^{\dagger T} \odot \boldsymbol{B}^{\dagger T})$ minimizes a transformed residual

$$\|(\boldsymbol{\mathcal{X}} - [\![\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}]\!])_{(1)}(\boldsymbol{C}^{\dagger T} \otimes \boldsymbol{B}^{\dagger T})\|_F$$

- Original motivation for the method came from collaboration on efficiently optimizing CP with generalized distance metrics with Ardavan Afshar, Cheng Qian, and Jimeng Sun[1]
- We may generally view the algorithm as performing alternating Mahalanobis distance minimization (AMDM)

$$f(\boldsymbol{A}^{(1)}, \cdots, \boldsymbol{A}^{(N)}) = \frac{1}{2}\|\boldsymbol{\mathcal{X}} - \boldsymbol{\mathcal{Y}}\|_M^2 = \frac{1}{2}\text{vec}(\boldsymbol{\mathcal{X}} - \boldsymbol{\mathcal{Y}})^T \boldsymbol{M} \text{vec}(\boldsymbol{\mathcal{X}} - \boldsymbol{\mathcal{Y}}), \quad (1)$$
$$\text{where } \boldsymbol{\mathcal{Y}} = [\![\boldsymbol{A}^{(1)}, \cdots, \boldsymbol{A}^{(N)}]\!].$$

with the ground metric $\boldsymbol{M}$ chosen at each subsweep as

$$\boldsymbol{M} = \left(\boldsymbol{M}^{(N)} \otimes \ldots \otimes \boldsymbol{M}^{(1)}\right)^{-1} \quad (2)$$
$$\text{where } \boldsymbol{M}^{(k)} = \boldsymbol{A}^{(k)}\boldsymbol{A}^{(k)T} + (\boldsymbol{I} - \boldsymbol{A}^{(k)}\boldsymbol{A}^{(k)\dagger}) \quad \forall k \in \{1, \ldots, N\}$$

[1] A. Ardavan, K. Yin, S. Yan, C. Qian, J.C. Ho, H. Park, and J. Sun, AAAI 2021.

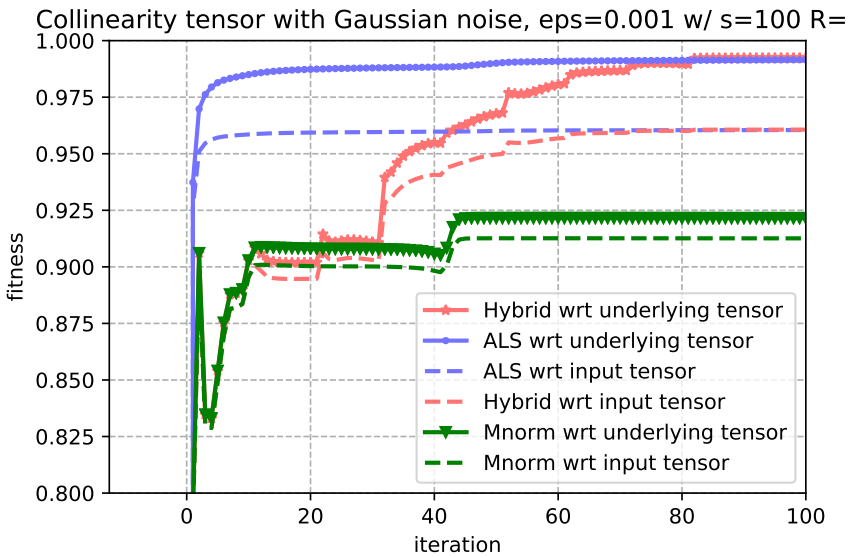
Hybrid Objective Optimization with AMDM

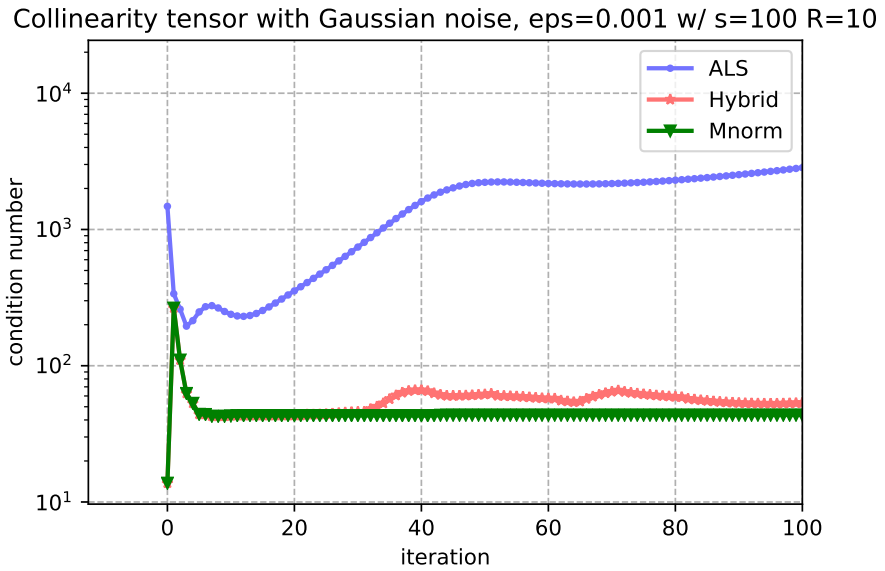
- With the ground metric $\boldsymbol{M} = \boldsymbol{I}$, we would reproduce the standard residual Frobenius norm
- Choose metric with $\boldsymbol{M}^{(n)} = \boldsymbol{U}^{(n)} \operatorname{diag}((\boldsymbol{s}_1^{(n)})^{-2}, \boldsymbol{1}) \boldsymbol{U}^{(n)T}$ where $\boldsymbol{A}^{(n)} = \boldsymbol{U}^{(n)} \operatorname{diag}(\boldsymbol{s}^{(n)}) \boldsymbol{V}^{(n)T}$ to interpolate between basic AMDM scheme and ALS
  - in computing $\boldsymbol{A}^{(n)\dagger}$ for each update, invert $R - D$ singular values $\boldsymbol{s}_1^{(n)}$
  - solve ALS-like reduced system related to remaining D singular values
- A generalized ground metric also allows us to extend the scheme to $R > s$

$$\begin{aligned} \boldsymbol{A}^{(n)} &= \boldsymbol{X}_{(n)} \boldsymbol{L}^{(n)} \boldsymbol{Z}^{(n)^{-1}}, \\ \text{where } \boldsymbol{L}^{(n)} &= \boldsymbol{L}_N \odot \ldots \odot \boldsymbol{L}_{n+1} \odot \boldsymbol{L}_{n-1} \odot \ldots \odot \boldsymbol{L}_1, \\ \text{and } \boldsymbol{Z}^{(n)} &= \boldsymbol{Z}_N * \ldots * \boldsymbol{Z}_{n+1} * \boldsymbol{Z}_{n-1} * \ldots * \boldsymbol{Z}_1, \end{aligned} \tag{3}$$

where each $\boldsymbol{L}_k = \boldsymbol{M}^{(k)} \boldsymbol{A}^{(k)}$ and $\boldsymbol{Z}_k = \boldsymbol{A}^{(k)T} \boldsymbol{M}^{(k)} \boldsymbol{A}^{(k)}$.

Approximate Decomposition Results with AMDM





- AMDM finds decomposition with lower CP condition number[1]
- Hybrid version gradually transitions from basic AMDM to ALS, achieving good fitness to underlying model

[1] P. Breiding and N. Vannieuwenhoven, SIMAX 2018.

Open Questions about AMDM

- Proof of local convergence to stationary points when R is less than the true rank
- Other views of the method in the approximate case (other than Mahalanobis distance minimization)
  - critical points of

$$\langle \boldsymbol{\mathcal{X}}, [\![\boldsymbol{U}^T, \boldsymbol{V}^T, \boldsymbol{W}^T]\!] \rangle - \text{tr}(\log(\boldsymbol{U}\boldsymbol{U}^T \otimes \boldsymbol{V}\boldsymbol{V}^T \otimes \boldsymbol{W}\boldsymbol{W}^T))$$

  - nested linear least squares solves
    1. solve for $\boldsymbol{Y}$ in $\boldsymbol{B}\boldsymbol{Y} \cong \boldsymbol{X}_{(2)}$
    2. unfold $\boldsymbol{Y}$ to third order tensor, then solve for $\boldsymbol{Z}$ in $\boldsymbol{C}\boldsymbol{Z} \cong Y_{(3)}$
    3. take the partial trace of $\boldsymbol{Z}$ to obtain updated $\boldsymbol{A}$
- Quantifying difference in conditioning of update rule (is $\boldsymbol{A}^{\dagger T} \odot \boldsymbol{B}^{\dagger T}$ a more well-conditioned left inverse of $\boldsymbol{A} \odot \boldsymbol{B}$ than its pseudoinverse?)